

Re: Penrose's Computing Pi Description?

Source: <http://sci.tech-archive.net/Archive/sci.logic/2005-06/msg00176.html>

- *From:* "Stephen Harris" <cyberguard-1048@xxxxxxxxxx>
 - *Date:* Thu, 09 Jun 2005 20:16:13 GMT
-

"H. J. Sander Bruggink" <sanderbruggink@xxxxxxxxxx> wrote in message
[news:d89bc0\\$71h\\$1@xx](mailto:news:d89bc0$71h$1@xx)

> Stephen Harris wrote:
>
> <snip penrose>
>
>> SH: I realize that computing infinity does not come to
>> an end or reach a completion. What does this have to
>> do with a condition of *not being allowed* for a TM?
>
> In the standard definition of a TM, it can only output things
> after it has finished. The "output" of a non-terminating TM
> is usually taken to be undefined.
>

Is that because it is assumed that the input is uncomputable?

> <snip penrose>
>
>> SH: I realize that expansions made earlier in the
>> computation will not have the same number of places
>> as some value reached later in the calculation, but I
>> wouldn't think this would be described as "subject
>> to possible change and so cannot be trusted."
>> Why can't it be trusted, how can it be *changed*?
>
> The way a TM output things, is by leaving the output on the
> tape. However, the tape is also where the TM stores
> intermediate results. Therefore, in principle, while the TM
> is running there is no way (in general) to distinguish the
> output from the intermediate results.
>
> Of course, for some particular TM, you could prove that
> some symbol on the tape will never change again. But in
> general, that is not true.
>

How about for Pi? How about for the infinite many of

Re: Penrose's Computing Pi Description?

irrationals produced by taking the square root of a prime?

>> I also don't fathom how his next method is
>> legitimate and excludes the possibility of "change"?
>
> The difference between the two methods is that the second
> stops for each digit of Pi, and therefore we can take the
> digit as output. The first never stops, and therefore, by
> definition, never outputs anything.
>
> Of course, in the first case, you have to "manually" run
> the TM for any natural number, so it is not really
> "computing Pi" on its own, IMHO.
>
> <snip>
>
>> Is there a flaw in how I understand this theory works?
>> Could someone please clarify a bit, Penrose's thinking.
>
> I hope this helps.
>
> groente
> -- Sander

Yes, thank you. I have one important question left, the other ones are minor. Is the algorithmic method which Penrose states can compute Pi (one digit at a time and then appending) is the algorithm the computation of one particular digit or is the algorithm considered to include the execution of the continuing series of computations.

Has the algorithm halted after finding one digit of Pi, or is the algorithm considered the entire process of finitely, step by step, churning out an endless sequence of digits? Are there two algorithms? Or one algorithm which includes writing finite values along the way. Or just one algorithm which writes a finite value and this algorithm is repeated over and over. Must an identical algorithm compute the same place in the decimal expansion of Pi. It seems to me that appending the last digit computed to the other digits already computed must be part of the algorithm and this part of the process doesn't halt so that the "entire" algorithm doesn't halt?

I hold the broader picture of what the algorithm includes, but I could be wrong.

Regards,
Stephen

- *Follow-Ups:*
 - ◆ *Re: Penrose's Computing Pi Description?*
 - ◇ *From:* H. J. Sander Bruggink

- *References:*
 - ◆ *Penrose's Computing Pi Description?*
 - ◇ *From:* Stephen Harris
 - ◆ *Re: Penrose's Computing Pi Description?*
 - ◇ *From:* H. J. Sander Bruggink

- Prev by Date: *Re: FOL, ZFC, NGB and Prolog*
- Next by Date: *Re: How do you pronounce Godel ??*
- Previous by thread: *Re: Penrose's Computing Pi Description?*
- Next by thread: *Re: Penrose's Computing Pi Description?*
- Index(es):
 - ◆ *Date*
 - ◆ *Thread*